SHIFT: RECEIVER-CONDITIONED PREDICTION-SPACE TRANSFER FOR LOW-DATA RGB-TO-THERMAL ADAPTATION

Anonymous authors
Paper under double-blind review

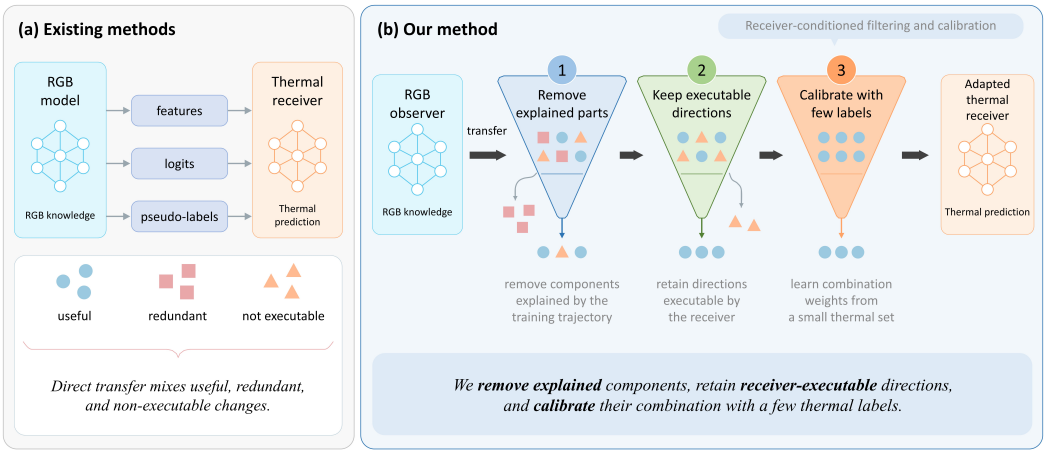

Figure 1: Motivation and overview of SHIFT. (a) Direct transfer of RGB features, logits, or pseudo-labels can mix useful, redundant, and non-executable changes. (b) In the task prediction space, SHIFT removes trajectory-explained components and retains locally receiver-executable directions. These directions are frozen before thermal support labels learn their signed weights, yielding an adapted model that uses thermal input alone.

ABSTRACT

Large-scale data and pretraining give RGB models rich semantic and geometric knowledge, whereas thermal perception remains limited by scarce and expensive task annotations. We study low-data RGB-to-thermal adaptation with a pretrained RGB model, abundant unlabeled RGB–thermal pairs, and only a small thermal support set, while keeping deployment thermal-only. Existing domain adaptation and cross-modal distillation methods transfer features, teacher predictions, or pseudo-labels, but they do not identify which transferred changes are new to the current thermal model and executable by its permitted update mechanism. We introduce SHIFT, a receiver-conditioned prediction-space transfer framework that selects RGB-induced changes that are novel relative to the native thermal trajectory and executable by the designated thermal receiver. Using predictive Fisher geometry, SHIFT removes task-invariant and trajectory-explained components, projects the remaining change onto the receiver's locally executable response space, and freezes a low-dimensional family of executable prediction directions. Thermal labels then estimate only their signed combination. On MSRS semantic segmentation, M3FD object detection, and MS2 monocular and stereo depth estimation, full SHIFT improves over four adaptation strategies initialized from the native thermal baseline under a 10% label budget. Shared-resource comparisons and 5/20-shot evaluations further support the value of receiver-conditioned filtering and low-dimensional calibration.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Large image collections, human annotations, and pretraining allow RGB vision models to acquire strong semantic and geometric capabilities for recognition, detection, and 3D perception (Gao et al., 2022; Lin et al., 2026; Wen et al., 2026). Thermal vision has much less high-quality data, and pixel labels, boxes, and depth annotations are particularly costly (Tang et al., 2022; Liu et al., 2022; Shin et al., 2023). Reusing capabilities already learned by RGB models is therefore an attractive way to reduce supervision for thermal tasks. This paper studies low-data RGB-to-thermal adaptation with a pretrained RGB model, abundant unlabeled RGB–thermal scene pairs, and a small set of thermal task labels, while retaining thermal-only inference at deployment.

Existing work addresses this setting through modality alignment and knowledge transfer. Domain adaptation narrows the RGB–thermal gap with shared features, pseudo-labels, or teacher updates (Gan et al., 2023; Do et al., 2024). Cross-modal distillation transfers RGB features or predictions and increasingly uses teacher design and sample selection to improve supervision quality (Gupta et al., 2016; Huo et al., 2024; Ranasinghe et al., 2026). Paired representation learning and infrared pretraining improve thermal initialization (Mao et al., 2025; Xing et al., 2026; Maheshwari et al., 2026), while low-rank updates and prompt modules reduce adaptation cost (Hu et al., 2022; Zhao et al., 2024). These approaches make RGB information available, yet the useful part of that information depends on the thermal model that receives it.

For an existing thermal model, task-irrelevant changes consume adaptation capacity, changes already represented by the model provide little new signal, and relevant changes may still be unreachable under a restricted update such as a prediction head or a LoRA module. The central question is thus not only how to transfer RGB knowledge, but which RGB-induced prediction changes are new relative to the current thermal model and can be realized by its allowed update mechanism. This receiver-conditioned view connects the content of transferred information to the function space available at the target.

SHIFT addresses this question in the target prediction space (Fig. 1). The RGB model acts as an observer and proposes changes relative to the current thermal prediction on paired scenes. A task-typing map puts the two modalities in a common output space. Predictive Fisher geometry then removes task-invariant degrees of freedom and the component explained by the native thermal training trajectory. The remaining change is projected onto the locally reachable thermal response space, producing an executable prediction direction. Paired data construct and freeze a family of such directions; the small thermal support set estimates their signed combination weights. RGB data therefore discover candidate directions, then the thermal model determines which directions it can realize, and task labels choose their final combination. This separation gives a low-dimensional target-side problem while preserving the full receiver implementation.

We evaluate SHIFT on MSRS semantic segmentation, M3FD object detection, and MS2 monocular and stereo depth estimation (Tang et al., 2022; Liu et al., 2022; Shin et al., 2023). The paper makes three contributions. First, it formulates transferable cross-modal information as a receiver-conditioned prediction change that is task-relevant, new relative to a native trajectory, and executable by the designated thermal receiver. Second, it introduces a Fisher-geometric construction that freezes these directions from unlabeled paired scenes and learns only their combination weights from a small thermal support set. Third, it gives a unified analysis of local direction selection, finite receiver updates, and the statistical complexity of the frozen prediction family, and validates the resulting framework across four thermal perception tasks.

2 RELATED WORK

2.1 RGB-TO-THERMAL ADAPTATION AND CROSS-MODAL TRANSFER

RGB-T analysis spans fusion, segmentation, detection, and tracking (Song et al., 2023). Fusion methods use both modalities at deployment (Tang et al., 2022; Liu et al., 2022; Zhao et al., 2024); SHIFT uses RGB during adaptation and deploys with thermal input alone. UMDA (Chen et al., 2026) combines RGB-T fusion, context attention, pseudo-labels, and self-distillation for segmentation without target annotations. Domain adaptation also uses invariant features (Ganin et al., 2016), domain-specific attention (Gan et al., 2023), and dual-domain teachers (Do et al., 2024). SAGA

reduces RGB color bias through semantic-aware grayscale augmentation for visible-to-thermal detection (D et al., 2025). These approaches improve cross-domain signals; SHIFT selects their prediction-space content according to the native thermal trajectory and executable response space.

Knowledge distillation transfers teacher responses (Hinton et al., 2015), including paired RGB supervision for other modalities (Gupta et al., 2016) and two-stream teachers for single-stream deployment (Zhang et al., 2022). C2KD uses bidirectional distillation and online sample selection (Huo et al., 2024). Thermal-Det, AnyThermal, UNIV, DuGI-MAE, and OpenRSS improve thermal supervision or initialization through RGB teachers, paired representations, or efficient updates (Ranasinghe et al., 2026; Maheshwari et al., 2026; Mao et al., 2025; Xing et al., 2026; Zhao et al., 2024). SHIFT adds receiver-conditioned direction selection and calibrates the resulting fixed family with few thermal labels.

2.2 Transferability estimation and low-dimensional adaptation

Transferability measures estimate how well a source model or representation matches a target task. LEEP uses source predictions on target data (Nguyen et al., 2020), and LogME evaluates the evidence of target labels given pretrained features (You et al., 2021). Complementary transfer studies also consider preserving capabilities already present in the student (Roth et al., 2024). SHIFT refines this idea from a model-level score to a set of target prediction directions: it removes trajectory-explained changes and keeps only the part that the receiver can express.

Parameter-efficient adaptation uses low-rank updates (Hu et al., 2022), visual prompts (Jia et al., 2022), or lightweight adapters (Chen et al., 2022). SEAR adapts a geometric transformer with LoRA for unpaired RGB-T reconstruction and also evaluates thermal-only inputs (Skorokhodov et al., 2026). It learns cross-modal geometric alignment; SHIFT selects trajectory-conditioned prediction directions within a receiver's allowed update space. Intrinsic-dimension studies examine low-dimensional parameter learning (Li et al., 2018), whereas the frozen prediction family in SHIFT makes target-side complexity depend on its calibration dimension (Bartlett & Mendelson, 2002).

3 Method

3.1 Problem setting and overview

For task t, O and T denote RGB and thermal inputs, respectively. The pretrained RGB model acts as the observer p_O and remains fixed during adaptation. Let the unlabeled paired set be $\mathcal{D}_P = \{(x_i^O, x_i^T)\}_{i=1}^N$ and the labeled thermal support set be $\mathcal{S} = \{(x_k^T, y_k)\}_{k=1}^n$, where $n \ll N$. We call the thermal model, its allowed update variables, and the update procedure the receiver. Let its parameters be θ_0, let $\rho_t : \Theta_R \to \mathcal{M}_t$ map receiver states to the task prediction space, and let $p_0 = \rho_t(\theta_0)$. The current thermal model and its native training trajectory are given resources; support labels are used only to learn the combination weights. We use SHIFT for the full construction, and distinguish its zero-weight endpoint, SHIFT(Native H), from the calibrated version, SHIFT (full).

Receiver–calibration separation. SHIFT first constructs the receiver-specific prediction family from the current thermal model, its native trajectory, and the unlabeled paired scenes. The support set then supplies task supervision for estimating the d-dimensional weight vector β within this fixed family. This ordering makes direction discovery and target calibration separate operations.

Figure 2 separates direction discovery from label calibration. We first extract a conditional change r from paired scenes, project it to an executable direction w, and assemble these directions into W. A fixed receiver realization turns W into a low-dimensional prediction family $\mathcal{G}_t$. The support set then estimates $\beta^\star$, yielding $q_E = \mathcal{G}_t(\beta^\star)$. Algorithm 1 summarizes the procedure.

3.2 Conditional information from the observer

We first place the two modalities in the same output type. The task-typing map $C_{O\to t,x_i}$ converts an RGB response into the thermal task object: corresponding pixel class probabilities for segmentation, objectness, class and box coordinates for detection, and depth or disparity at corresponding locations

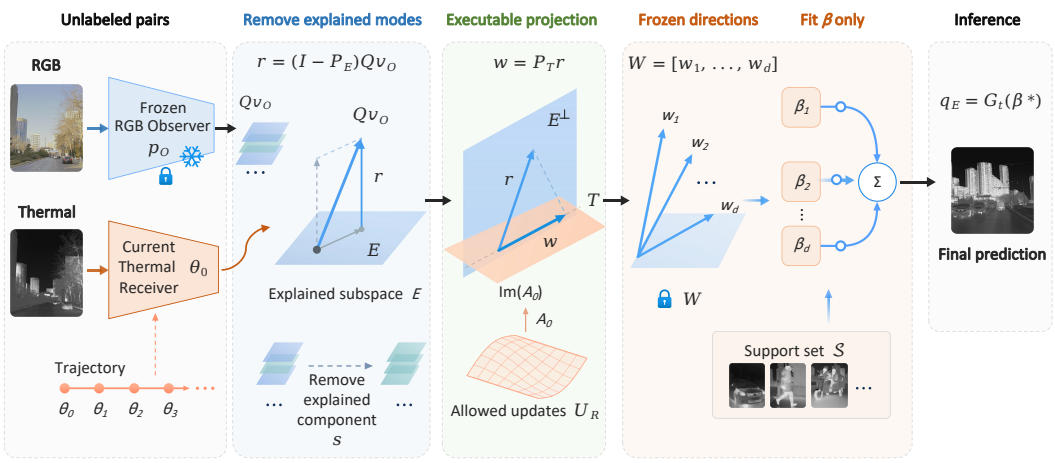

Figure 2: SHIFT. Paired RGB and thermal predictions are mapped to a common task space. The native thermal trajectory removes explained changes, and the receiver response space keeps executable directions. The resulting directions form a fixed W; a small thermal support set learns β, and $\mathcal{G}_t(\beta^\star)$ produces thermal-only predictions.

for geometry. Discrete outputs use probability coordinates; continuous quantities use fixed-covariance Gaussian location coordinates.

We compare local prediction changes with the predictive Fisher metric (Martens, 2020). For tangent vectors $a, b \in T_p\mathcal{M}_t$, define

$$g_{t,p}(a,b) = \int_{\mathcal{X}_t} g^F_{t,x,p(x)}(a(x), b(x))\ \mathrm{d}\mu_t(x), \tag{1}$$

where $\mathcal{X}_t$ is the thermal input domain, g^F is the Fisher inner product of one output distribution, and μ_t weights inputs and valid prediction locations. This metric is the quadratic term of the local KL divergence. We estimate inner products and projections by the weighted average over paired scenes; Appendix A.1 gives the explicit forms.

The task map and a local logarithm give the observer's change:

$$\widetilde{p}_O(x_i^T) = C_{O\to t,x_i}[p_O(x_i^O)], \qquad v_O(x_i^T) = \mathrm{Log}_{p_0(x_i^T)}\widetilde{p}_O(x_i^T). \tag{2}$$

The logarithm represents the difference at a common base point, where it can be added and projected. For fixed-covariance Gaussian coordinates it is the coordinate difference. For class probabilities, we use the square-root sphere and map the resulting logarithm back to probability coordinates.

Task evaluation may identify multiple predictions as equivalent. Let $\pi_t : \mathcal{M}_t \to \mathcal{Q}_t$ be the corresponding quotient map, define $q_0 = \pi_t(p_0)$, $\mathcal{H} = T_{q_0}\mathcal{Q}_t$, and $Q = D\pi_t|_{p_0}$. We represent $\mathcal{H}$ by the Fisher orthogonal complement of the removed directions and denote its inner product by g_0. For scale-invariant monocular depth, Q removes the constant log-depth direction; segmentation, detection, and metric stereo disparity use $Q = I$.

The native thermal trajectory provides a reference for changes already represented by the model. We map selected checkpoints to the same tangent space, take adjacent differences, and apply Q, obtaining $e_1, \ldots, e_m \in \mathcal{H}$. Their span is $E = \mathrm{span}\{e_1, \ldots, e_m\}$. Let P_E be the g_0-orthogonal projection onto E. The conditional observer change is

$$r = (I - P_E)Qv_O. \tag{3}$$

Thus r captures novelty relative to the selected trajectory: it is the component orthogonal to the span of native prediction changes. Appendix A.2 proves the associated best-approximation property.

3.3 Executable Prediction Directions

The receiver update domain determines which prediction changes can be realized. Let $U_R \subseteq T_{\theta_0}\Theta_R$ be the span of selected directions in the allowed parameter domain, which may include a prediction

head, LoRA modules (Hu et al., 2022), or other specified parameters. The receiver derivative is

$$A_0 = Q\, D\rho_t(\theta_0)|_{U_R} : U_R \longrightarrow \mathcal{H}. \tag{4}$$

For $\delta \in U_R$, $A_0\delta$ is the first-order prediction response. We obtain response columns with Jacobian–vector products (JVPs), without materializing the full Jacobian.

We enforce reachability and non-redundancy together. Requiring $P_E A_0 \delta = 0$ gives

$$\mathcal{T} = A_0 \ker(P_E A_0) = \mathrm{Im}(A_0) \cap E^{\perp}. \tag{5}$$

Every direction in $\mathcal{T}$ is both a receiver response and orthogonal to the trajectory reference. If $P_{\mathcal{T}}$ is the g_0-orthogonal projection, the executable prediction direction is

$$w = P_{\mathcal{T}} r = \arg\min_{z \in \mathcal{T}} \|r - z\|_{g_0}^2 . \tag{6}$$

The projection selects the executable change closest to the conditional observer change. In practice, we compute the null space of $P_E A_0$ in the response coefficients and solve a Fisher-weighted least-squares problem. Since $\mathcal{T} \subseteq E^{\perp}$, the same result is $w = P_{\mathcal{T}} Q v_O$. The three components $P_E Q v_O$, w, and $r - w$ are pairwise orthogonal; a proof appears in Appendix A.3.

3.4 A FROZEN LOW-DIMENSIONAL PREDICTION FAMILY

For d fixed observer candidates $v_O^{(1)}, \ldots, v_O^{(d)}$, we repeat the construction above and obtain $w_1, \ldots, w_d$. The response construction rules are fixed before support labels are read. Stacking the directions gives

$$W = [w_1, \ldots, w_d] : \mathbb{R}^d \to \mathcal{T}, \qquad u_\beta = W\beta = \sum_{j=1}^{d} \beta_j w_j. \tag{7}$$

The sign of β_j selects a direction or its reverse, and its magnitude controls the strength.

The projection also provides an implementation coefficient for each column. We insert these coefficients into a fixed receiver update procedure and obtain a prediction function that can be evaluated on new thermal inputs. Parameterized receivers inject the coefficients into parameters or the native loss; function receivers combine fixed thermal prediction functions. After fixing the implementation rules, data, schedule, and randomness, the weights define

$$q_\beta = \mathcal{G}_t(\beta), \qquad \mathcal{G}_t : \mathbb{R}^d \to \mathcal{Q}_t, \qquad q_H = \mathcal{G}_t(0). \tag{8}$$

Here q_H is the zero-weight endpoint, or SHIFT(Native H), and q_β is the final prediction associated with β. W describes the local directions at the base point, whereas $\mathcal{G}_t$ includes their receiver realization and subsequent finite updates. Appendix A.5 relates the two through a first-order response and a second-order nonlinearity bound.

3.5 LEARNING COMBINATION WEIGHTS FROM FEW LABELS

The support labels determine how the fixed directions should be combined. Let $R_{\mathcal{S}}$ evaluate a prediction or derivative on support inputs, and let $\boldsymbol{e}_j$ be the j-th standard coordinate vector. The endpoint response columns are

$$B_j = R_{\mathcal{S}} D\mathcal{G}_t(0)[\boldsymbol{e}_j], \qquad B\beta = \sum_{j=1}^{d} \beta_j B_j. \tag{9}$$

These columns describe the support-set prediction change produced by the complete receiver realization near the zero-weight endpoint. We compute them by propagating JVPs through the differentiable update steps.

Let $L_{\mathcal{S}}$ be the support loss in the same quotient coordinates, and let $\widehat{g}_{\mathcal{S}}$ be the empirical Fisher inner product at q_H. Define

$$\begin{aligned} y_{\mathcal{S}} &= -\operatorname{grad}_{\widehat{g}_{\mathcal{S}}} L_{\mathcal{S}}(q_H), \\ G_{ij} &= \langle B_i, B_j \rangle_{\widehat{g}_{\mathcal{S}}}, \\ c_i &= \langle B_i, y_{\mathcal{S}} \rangle_{\widehat{g}_{\mathcal{S}}} = -\left. \frac{\partial L_{\mathcal{S}}(\mathcal{G}_t(\beta))}{\partial \beta_i} \right|_{\beta=0}. \end{aligned} \tag{10}$$

The vector $y_{\mathcal{S}}$ is the local loss-decreasing direction, G is the response Gram matrix, and c measures each response column's alignment with the supervised decrease.

We combine the first-order loss decrease with a predictive Fisher cost:

$$m(\beta) = -c^\top \beta + \frac{1}{2}\beta^\top G\beta + \frac{1}{2}\beta^\top M_{\mathrm{coord}}\beta, \qquad M_{\mathrm{coord}} \succ 0. \tag{11}$$

The matrix M_{coord} controls the weight scale using a fixed response normalization rule. Since $\|B\beta\|^2_{\widehat{g}_{\mathcal{S}}} = \beta^\top G\beta$ and $\langle B\beta, y_{\mathcal{S}}\rangle_{\widehat{g}_{\mathcal{S}}} = c^\top\beta$, the solution is

$$\begin{aligned}\beta^\star &= \arg\min_{\beta\in\mathbb{R}^d}\left\{\frac{1}{2}\|B\beta - y_{\mathcal{S}}\|^2_{\widehat{g}_{\mathcal{S}}} + \frac{1}{2}\beta^\top M_{\mathrm{coord}}\beta\right\} \\ &= (G + M_{\mathrm{coord}})^{-1}c.\end{aligned} \tag{12}$$

The labels enter a d-dimensional linear system through c, while B, G, and the coordinate regularizer are fixed by the support inputs and the frozen prediction family. The final prediction is

$$q_E = \mathcal{G}_t(\beta^\star). \tag{13}$$

This is SHIFT (full), and q_H is its matched zero-weight reference. With no calibration labels, $\beta = 0$ and $q_E = q_H$. Uniqueness and finite-update conditions appear in Appendix A.4 and Appendix A.6; the label complexity of the frozen family is given in Appendix A.7.

3.6 Training and deployment

Algorithm 1 follows the four stages above. Paired scenes construct and freeze the prediction family and its receiver realization. Thermal labels estimate the combination weights, and deployment keeps only the resulting thermal model or prediction function.

Algorithm 1 SHIFT: direction filtering, weight learning, and thermal-only deployment

Input: Paired scenes $\mathcal{D}_P$, support set $\mathcal{S}$, observer p_O, current model θ_0 and native trajectory; maps $C_{O\to t}, Q$, update domain U_R, implementation rule and schedule; d candidate construction rules and coordinate regularizer
Output: Thermal-only prediction function q_E

Phase I: construct the prediction family from paired scenes
Compute p_0, q_0, g_0, E and the receiver response A_0
Construct $\mathcal{T} = \mathrm{Im}(A_0) \cap E^\perp$
for $j = 1, \ldots, d$ **do**
    Construct $v_O^{(j)}$ on paired scenes
    $r_j \leftarrow (I - P_E)Qv_O^{(j)}$
    $w_j \leftarrow P_{\mathcal{T}} r_j$
end for
Freeze $W = [w_1, \ldots, w_d]$, column implementations, and $\mathcal{G}_t$
Phase II: learn the combination weights from thermal labels
if $\mathcal{S} = \varnothing$ **then**
    return $\mathcal{G}_t(0)$ ▷ SHIFT(Native H)
end if
$q_H \leftarrow \mathcal{G}_t(0)$; compute $B, G, M_{\mathrm{coord}} \succ 0$ from support inputs
Compute $c = -\nabla_\beta L_{\mathcal{S}}(\mathcal{G}_t(\beta))|_{\beta=0}$ from support labels
Solve $(G + M_{\mathrm{coord}})\beta^\star = c$
return $q_E = \mathcal{G}_t(\beta^\star)$

4 Experiments

4.1 Tasks and task-specific receivers

We evaluate SHIFT on MSRS semantic segmentation, M3FD object detection, and MS2 monocular and stereo depth estimation (Tang et al., 2022; Liu et al., 2022; Shin et al., 2023). All deployed models use thermal input only. MSRS represents nine-class pixel probabilities and realizes the

Table 1: Recognition results with a 10% thermal-label budget. The four adaptation baselines start from Native-H, also reported as SHIFT(Native H). All metrics are percentages.

Method	MSRS	M3FD		
	mIoU ↑	AP ↑	AP50 ↑	AP75 ↑
SHIFT(Native H)	53.86	29.64	51.02	31.32
Native-H + Head	56.21	31.08	53.11	32.66
Native-H + SSF	56.81	31.73	54.27	33.40
Native-H + LoRA-r8	57.56	32.82	55.76	34.69
Native-H + Full-FT	58.33	34.00	57.26	35.68
SHIFT (full)	61.30	35.99	60.34	37.67

Table 2: Depth results with the same Native-H initialization and label budget as Table 1. Mono uses per-image median scaling; Stereo uses metric depth. D1 counts pixels with disparity error above 3 px and 5%; δ_1 is the fraction with depth error ratio below 1.25.

Method	MS2 Mono		MS2 Stereo			
	SI-AbsRel ↓	SI-δ_1 ↑	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
SHIFT(Native H)	0.1302	0.8253	0.4023	0.39	0.0833	0.9328
Native-H + Head	0.1282	0.8347	0.3985	0.38	0.0826	0.9339
Native-H + SSF	0.1264	0.8411	0.3978	0.37	0.0824	0.9345
Native-H + LoRA-r8	0.1254	0.8466	0.3959	0.37	0.0821	0.9346
Native-H + Full-FT	0.1314	0.8190	0.3918	0.36	0.0816	0.9359
SHIFT (full)	0.1217	0.8634	0.3879	0.35	0.0807	0.9375

selected prediction directions with LoRA and the segmentation head. M3FD represents objectness, class, and normalized box coordinates and uses class-conditioned detector responses. MS2 monocular depth uses scale-quotient log-depth coordinates and a fixed prediction-function family. MS2 stereo depth preserves metric scale and realizes the selected prediction directions through the native FastFoundationStereo refinement process (Wen et al., 2026). The response basis follows the structure of each receiver. MSRS uses 53 predefined LoRA/head groups; M3FD uses six foreground-class directions and one background direction, with proposal indices shared across response evaluations. Mono uses nine fixed prediction functions, and Stereo uses eight recurrent-stage response directions. These probes are constructed before calibration labels are read. Their count determines response computation, whereas the learned weight vectors have 9, 7, 9, and 8 coordinates, respectively. Linear dependence further reduces the effective endpoint ranks to 9, 3, 5, and 4. This distinction connects the large receiver update domains to the small estimation problems solved with thermal labels; construction details appear in Appendix B.2. Throughout the experiments, recognition scores are percentages, EPE is reported in pixels, D1 is reported as a percentage, and SI-AbsRel and AbsRel are dimensionless errors. SI-δ_1 and δ_1 are reported as fractions in $[0, 1]$.

4.2 Low-data RGB-to-thermal adaptation

Our main comparison starts the four standard adaptation strategies from Native-H, the thermal baseline denoted by SHIFT(Native H) and $q_H = \mathcal{G}_t(0)$. With the same thermal label budget, Native-H + Head trains the task head, Native-H + SSF (Lian et al., 2022) trains feature scales, shifts, and the head, Native-H + LoRA-r8 (Hu et al., 2022) trains rank-eight modules and the head, and Native-H + Full-FT updates all parameters. SHIFT (full) calibrates its fixed prediction family relative to the same native baseline. This comparison tests whether selecting and combining RGB-induced directions improves on further supervised adaptation of an already capable thermal model. Pair-pretrained comparisons are reported in Appendix C.1.

Tables 1 and 2 show that SHIFT (full) improves over all four Native-H-initialized adapters. Against the strongest adapter for each task, it gains 2.97 mIoU on MSRS and 1.99 AP on M3FD, and reduces Mono SI-AbsRel from 0.1254 to 0.1217 and Stereo EPE from 0.3918 to 0.3879. Full-FT is the

Table 3: Adaptation from Native-H with 5 and 20 thermal labels. The native baseline is fixed across budgets. Full detection and depth metrics appear in Appendix C.2.

Labels	Method	MSRS mIoU ↑	M3FD AP ↑	MS2 Mono SI-AbsRel ↓	MS2 Stereo EPE (px) ↓
5	Native-H + Head	54.12	29.86	0.1301	0.4016
	Native-H + SSF	54.27	29.95	0.1300	0.4012
	Native-H + LoRA-r8	54.29	30.05	0.1301	0.4014
	Native-H + Full-FT	54.36	30.20	0.1329	0.3995
	SHIFT (full)	56.47	31.78	0.1296	0.3961
20	Native-H + Head	54.78	30.55	0.1281	0.4009
	Native-H + SSF	55.36	30.66	0.1289	0.3993
	Native-H + LoRA-r8	55.51	31.56	0.1295	0.3995
	Native-H + Full-FT	55.60	32.20	0.1322	0.3959
	SHIFT (full)	59.36	35.03	0.1246	0.3921

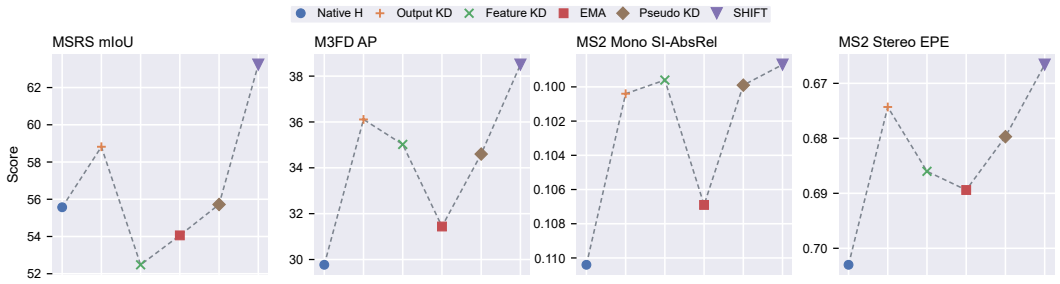

Figure 3: Shared-resource transfer-objective comparison with pooled out-of-fold evaluation. Each panel reports one task metric; higher is better for MSRS and M3FD, and lower is better for MS2 Mono and Stereo. Full values are given in Appendix B.

strongest standard adapter for recognition and stereo, whereas LoRA-r8 performs best on Mono. SHIFT improves across these different update regimes by using the labels to combine selected prediction changes. We also compare adaptation with 5 and 20 labeled thermal examples, keeping Native-H and the paired resources fixed. SHIFT improves over the strongest Native-H adapter on all four tasks at both budgets (Table 3). Its recognition gains grow from 2.11 mIoU and 1.58 AP with five labels to 3.76 mIoU and 2.83 AP with twenty. Mono SI-AbsRel and Stereo EPE also decrease as more labels refine the combination weights, consistent with the role of supervision in the fixed family.

4.3 Shared-resource transfer objectives

We fix the RGB observer, native thermal model H, unlabeled paired scenes, label pool, task readout, receiver, and update budget, and vary only the cross-modal transfer objective. The comparison uses two-fold pooled out-of-fold evaluation. Direct output KD transfers the observer's task output; Feature KD distills paired intermediate features; EMA consistency constrains predictions with an exponential-moving-average teacher; confidence pseudo-label KD uses teacher predictions above a fixed confidence threshold. These controls use the same resources as SHIFT and provide a direct comparison of transfer-objective design.

SHIFT (full) is the strongest objective across the four panels. It improves over the strongest alternative by 4.41 mIoU on MSRS, 2.39 AP on M3FD, 0.0009 SI-AbsRel on monocular depth, and 0.0077 EPE on stereo depth. The result shows that the shared RGB signal becomes more useful when the transfer objective is conditioned on the native trajectory and the receiver response space.

4.4 Mechanism evidence

We next test whether the result depends on the content of the selected directions rather than only their number or Fisher energy. Sample-shuffled preserves the observer response statistics while breaking

Table 4: Matched direction controls. Shuffled and Isometric match SHIFT in direction count, weight learning, and Fisher prediction energy. Mono uses support OOF; the other tasks use their main evaluation sets. Additional metrics are in Table 10.

Method	MSRS mIoU ↑	M3FD AP ↑	MS2 Mono SI-AbsRel ↓	MS2 Stereo EPE (px) ↓
SHIFT(Native H)	53.86	29.64	0.0465	0.4023
Shuffled	55.85	32.03	0.0454	0.3926
Isometric	55.65	32.59	0.0457	0.3936
SHIFT (full)	61.30	35.99	0.0444	0.3879

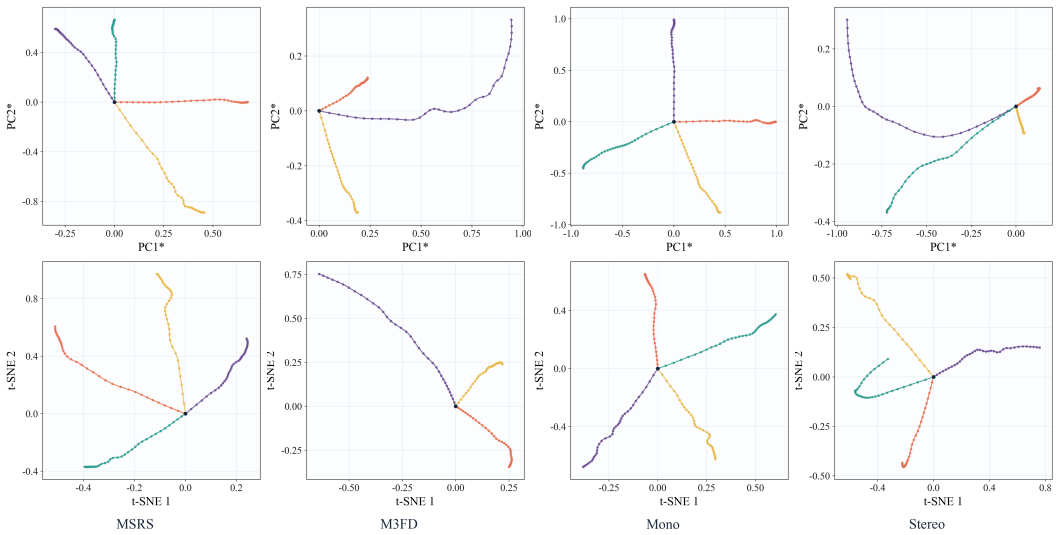

Figure 4: Receiver-state geometry for MSRS, M3FD, MS2 Mono, and MS2 Stereo. The upper row shows relative parameter displacement from a shared anchor; the lower row shows a t-SNE embedding of the same states. Colors identify receiver regions within each task.

scene correspondence; Sign-isometric applies a fixed equal-energy perturbation. Both use the same weight estimator and receiver as SHIFT. Table 4 shows that SHIFT achieves the strongest result on all four tasks under matched direction count and energy. Additional metrics and task-specific controls appear in Appendix B.

The component controls in Table 11 connect these gains to specific design choices. On M3FD, removing trajectory conditioning reduces AP from 35.99 to 30.88, and removing receiver projection reduces it to 35.09. Applying the selected change only at the output gives 32.12 AP, showing the importance of realizing the direction through the detector. On Mono, alternative scale coordinates increase SI-AbsRel from 0.1217 to 0.1280, while fixed combination weights give 0.1225. Together, these comparisons link performance to trajectory-based filtering, receiver realization, and task-consistent calibration within the fixed prediction family.

Figure 4 complements these controls with receiver-state visualizations. The upper row projects parameter displacements relative to a shared anchor; the lower row uses t-SNE to display local neighborhoods among the same states. The colored branches summarize how different receiver regions evolve within each task.

5 Conclusion

SHIFT selects RGB-induced prediction changes relative to a thermal model's native trajectory and executable response space. Unlabeled paired scenes define a fixed prediction family, while few thermal labels learn its combination weights. Experiments across four tasks support this separation of direction discovery and calibration, enabling low-data adaptation with thermal-only inference.

AI USE STATEMENT

Generative AI tools assist with English editing for clarity and readability and with mathematical derivations, including checks of intermediate algebraic and logical steps and suggestions for derivation steps or corrections. The authors independently review and verify all AI-assisted mathematical content before its inclusion and take full responsibility for the final paper.

REFERENCES

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16664–16678, 2022.

Zeyang Chen, Chunyu Lin, Yao Zhao, and Tammam Tillo. Unsupervised multi-modal domain adaptation for RGB-T semantic segmentation. *Computer Vision and Image Understanding*, 263: 104573, 2026. doi: 10.1016/j.cviu.2025.104573.

Manjunath D, Aniruddh Sikdar, Prajwal Gurunath, Sumanth Udupa, and Suresh Sundaram. SAGA: Semantic-aware gray color augmentation for visible-to-thermal domain adaptation across multi-view drone and ground-based vision systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 4617–4627, 2025.

Dinh Phat Do, Taehoon Kim, Jaemin Na, Jiwon Kim, Keonho Lee, Kyunghwan Cho, and Wonjun Hwang. D3T: Distinctive dual-domain teacher zigzagging across RGB-thermal gap for domain-adaptive object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23313–23322, 2024.

Lu Gan, Connor Lee, and Soon-Jo Chung. Unsupervised RGB-to-thermal domain adaptation via multi-domain attention network. In *IEEE International Conference on Robotics and Automation*, pp. 6014–6020, 2023.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(59):1–35, 2016.

Peng Gao, Teli Ma, Hongsheng Li, Ziyi Lin, Jifeng Dai, and Yu Qiao. MCMAE: Masked convolution meets masked autoencoders. In *Advances in Neural Information Processing Systems*, 2022.

Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross modal distillation for supervision transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2827–2836, 2016. URL https://openaccess.thecvf.com/content_cvpr_2016/html/Gupta_Cross_Modal_Distillation_CVPR_2016_paper.html.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Fushuo Huo, Wenchao Xu, Jingcai Guo, Haozhao Wang, and Song Guo. C2KD: Bridging the modality gap for cross-modal knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16006–16015, 2024.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727, 2022.

Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. In *International Conference on Learning Representations*, 2018.

Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://arxiv.org/abs/2210.08823.

Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Yang Zhao, Sida Peng, Hengkai Guo, Xiaowei Zhou, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. In *International Conference on Learning Representations*, 2026. Oral presentation.

Jinyuan Liu, Xin Fan, Zhanbo Huang, Guanyao Wu, Risheng Liu, Wei Zhong, and Zhongxuan Luo. Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5802–5811, 2022.

Parv Maheshwari, Jay Karhade, Yogesh Chawla, Isaiah Adu, Florian Heisen, Andrew Porco, Andrew Jong, Yifei Liu, Santosh Pitla, Sebastian Scherer, and Wenshan Wang. AnyThermal: Towards learning universal representations for thermal perception. In *IEEE International Conference on Robotics and Automation*, 2026.

Fangyuan Mao, Shuo Wang, Jilin Mei, Shun Lu, Chen Min, Fuyang Liu, Xiaokun Feng, Meiqi Wu, and Yu Hu. UNIV: Unified foundation model for infrared and visible modalities. *arXiv preprint arXiv:2509.15642*, 2025. URL https://arxiv.org/abs/2509.15642.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Cuong Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. LEEP: A new measure to evaluate transferability of learned representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7294–7305, 2020.

Yasiru Ranasinghe, Elim Schenck, Florence Yellin, Shuowen Hu, Christopher Funk, and Vishal M. Patel. Thermal-det: Language-guided cross-modal distillation for open-vocabulary thermal object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 34628–34637, 2026.

Karsten Roth, Lukas Thede, A. Sophia Koepke, Oriol Vinyals, Olivier Hénaff, and Zeynep Akata. Fantastic gains and where to find them: On the existence and prospect of general knowledge transfer between any pretrained model. In *International Conference on Learning Representations*, 2024.

Ukcheol Shin, Jinsun Park, and In So Kweon. Deep depth estimation from thermal image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1043–1053, 2023.

Vsevolod Skorokhodov, Chenghao Xu, Shuo Sun, Olga Fink, and Malcolm Mielle. SEAR: Simple and efficient adaptation of visual geometric transformers for unpaired RGB+Thermal 3D reconstruction. In *European Conference on Computer Vision*, 2026. URL https://arxiv.org/abs/2603.18774.

Kechen Song, Ying Zhao, Liming Huang, Yunhui Yan, and Qinggang Meng. RGB-T image analysis technology and application: A survey. *Engineering Applications of Artificial Intelligence*, 120: 105919, 2023. doi: 10.1016/j.engappai.2023.105919.

Linfeng Tang, Jiteng Yuan, Hao Zhang, Xingyu Jiang, and Jiayi Ma. PIAFusion: A progressive infrared and visible image fusion network based on illumination aware. *Information Fusion*, 83–84: 79–92, 2022.

Bowen Wen, Shaurya Dewan, and Stan Birchfield. Fast-foundationstereo: Real-time zero-shot stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7513–7524, 2026.

Yinghui Xing, Xiaoting Su, Shizhou Zhang, Donghao Chu, and Di Xu. DuGI-MAE: Improving infrared mask autoencoders via dual-domain guidance. In *Proceedings of the AAAI Conference on Artificial Intelligence, 40(32)*, pp. 27135–27143, 2026.

Kaichao You, Yong Liu, Jianmin Wang, and Mingsheng Long. LogME: Practical assessment of pre-trained models for transfer learning. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12133–12143, 2021.

Heng Zhang, Elisa Fromont, Sébastien Lefèvre, and Bruno Avignon. Low-cost multispectral scene analysis with modality distillation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 803–812, 2022.

Guoqiang Zhao, Junjie Huang, Xiaoyun Yan, Zhaojing Wang, Junwei Tang, Yangjun Ou, Xinrong Hu, and Tao Peng. Open-vocabulary RGB-thermal semantic segmentation. In *European Conference on Computer Vision*, pp. 304–320, 2024.

A SUPPLEMENTARY THEORY AND PROOFS

This appendix analyzes the information decomposition, executable directions, weight estimation, and generalization of SHIFT. The derivations use three objects: the directions W selected at the base point, the first-order response $D\mathcal{G}_t(0)$ of the complete adaptation process, and the finite prediction $\mathcal{G}_t(\beta)$. Trajectory projection and receiver constraints determine W, endpoint responses determine the weights, and finite-update bounds connect these objects to task loss. We then derive the label complexity of the frozen prediction family.

A.1 PREDICTIVE GEOMETRY AND TASK-INVARIANT COORDINATES

The predictive Fisher metric used in the paper has explicit forms for the two output types. For a categorical probability $p = (p_1, \dots, p_C)$ with $p_c > 0$, tangent vectors satisfy $\sum_c a_c = \sum_c b_c = 0$, and

$$\begin{aligned} g_p^F(a,b) &= \sum_{c=1}^{C} \frac{a_c b_c}{p_c}, \\ \mathrm{KL}(p \,\|\, p + \tau a) &= \frac{\tau^2}{2} g_p^F(a,a) + O(\tau^3). \end{aligned} \tag{A.1}$$

The second identity follows by expanding $-\sum_c p_c \log(1 + \tau a_c / p_c)$; the first-order term vanishes because $\sum_c a_c = 0$. The map $p \mapsto 2\sqrt{p}$ sends a to $a/\sqrt{p}$, whose Euclidean inner product is g_p^F. Categorical logarithms can therefore be evaluated on the sphere of radius two and mapped back to probability coordinates.

For the fixed-covariance Gaussian location model $\mathcal{N}(z, \Sigma)$, with $\Sigma \succ 0$, the Fisher inner product in location coordinates is

$$g_z^F(a,b) = a^\top \Sigma^{-1} b. \tag{A.2}$$

The score of the location parameter is $\Sigma^{-1}(Y - z)$, whose second moment is Σ^{-1}. Summing these inner products with the input and valid-location weights in the paper gives Eq. (1). Thus fixed-covariance continuous outputs use a weighted Euclidean geometry.

The quotient coordinates follow from the invariances specified by the task. For scale-invariant monocular depth, global scale is the constant vector $\mathbf{1}$ in log-depth coordinates. If $\Lambda \succ 0$ is the Fisher weight matrix over valid locations, the orthogonal complement of this invariant direction is

$$Qa = a - \mathbf{1} \frac{\mathbf{1}^\top \Lambda a}{\mathbf{1}^\top \Lambda \mathbf{1}}. \tag{A.3}$$

Differentiating $\|a - s\mathbf{1}\|_\Lambda^2$ with respect to s gives this coefficient and $\mathbf{1}^\top \Lambda Q a = 0$. This is the weighted projection that removes per-image scale. Segmentation, detection, and metric stereo disparity use $Q = I$.

A.2 THE NATIVE TRAJECTORY AS A COMPARISON REFERENCE

The native trajectory supplies a computable linear reference for the part of a change already represented by the current model. In the quotient tangent space $(\mathcal{H}, g_0)$, let $v = Qv_O$, and let $E = \mathrm{span}\{e_1, \dots, e_m\}$ be the span of selected trajectory changes. Here, trajectory-explained means linearly representable by these changes, and the conditional change is its orthogonal complement. The trajectory, task coordinates, and Fisher metric together define this decomposition; the task loss decides how the resulting directions are used.

All projections use the same inputs, valid locations, and weights as direction discovery. The empirical Fisher space used in the implementation is finite-dimensional. In the Hilbert-space formulation, the finite trajectory set and finite update domain make E and $\mathrm{Im}(A_0)$ closed subspaces, so the orthogonal projections are well defined.

Proposition A.1 (Best linear decomposition relative to the trajectory)**.** *For any $v \in \mathcal{H}$, $P_E v$ is the unique best approximation to v in E, and $r = (I - P_E)v$ satisfies*

$$\begin{aligned} P_E v &= \arg\min_{e \in E} \|v - e\|_{g_0}^2, \qquad r \perp E, \\ \|v\|_{g_0}^2 &= \|P_E v\|_{g_0}^2 + \|r\|_{g_0}^2. \end{aligned} \tag{A.4}$$

Replacing the trajectory basis does not change $P_E v$ or r when the span is unchanged.

Proof. For $e \in E$, $v - e = r + (P_E v - e)$, and the two terms are orthogonal. Hence

$$\|v - e\|_{g_0}^2 = \|r\|_{g_0}^2 + \|P_E v - e\|_{g_0}^2 .$$

The right-hand side is minimized exactly at $e = P_E v$. Setting $e = 0$ gives the energy decomposition. Orthogonal projection is uniquely determined by the subspace and inner product, so it is independent of the chosen basis. □

Direction construction and weight learning have different roles. The former extracts changes outside the trajectory reference, whereas the latter chooses their direction and strength. Consider a Euclidean prediction space with $q_0 = 0$, $E = \mathrm{span}\{\boldsymbol{e}_1\}$, $A_0 = I$, and $v = r = w = \boldsymbol{e}_2$. For targets $y_\pm = \pm\boldsymbol{e}_2$, the squared loss $\ell_y(q) = \frac{1}{2}\|q - y\|_2^2$ satisfies

$$\ell_{y_\pm}(\alpha w) - \ell_{y_\pm}(0) = \frac{1}{2}\alpha^2 \mp \alpha. \tag{A.5}$$

The same trajectory reference and executable direction therefore have opposite useful signs for the two targets. This is why SHIFT fixes candidate directions from paired scenes and learns signed weights from task labels. The labels in the motivation figure correspond to $P_E v$ and r.

A.3 Constrained projection to executable directions

Executability requires a selected direction to come from an allowed receiver update, while non-redundancy requires it to be orthogonal to the trajectory reference. Let $A_0 : U_R \to \mathcal{H}$ be the receiver derivative from the paper.

Proposition A.2 (Conditional reachable space and orthogonal decomposition)**.** *The constrained reachable space is*

$$\mathcal{T} = A_0 \ker(P_E A_0) = \mathrm{Im}(A_0) \cap E^\perp. \tag{A.6}$$

For $r = (I - P_E)v$, the unique best executable direction is

$$w = P_{\mathcal{T}} r = P_{\mathcal{T}} v = \operatorname*{arg\,min}_{z \in \mathcal{T}} \|r - z\|_{g_0}^2 . \tag{A.7}$$

Moreover, $v = P_E v + w + (r - w)$ is a pairwise orthogonal decomposition and

$$\|v\|_{g_0}^2 = \|P_E v\|_{g_0}^2 + \|w\|_{g_0}^2 + \|r - w\|_{g_0}^2 . \tag{A.8}$$

Proof. If $z = A_0\delta$ with $\delta \in \ker(P_E A_0)$, then $z \in \mathrm{Im}(A_0)$ and $P_E z = 0$, so $z \in E^\perp$. Conversely, if $z \in \mathrm{Im}(A_0) \cap E^\perp$, choose $\delta \in U_R$ with $z = A_0\delta$. Then $P_E A_0 \delta = P_E z = 0$, proving Eq. (A.6).

Because $\mathcal{T} \subseteq E^\perp$, $P_{\mathcal{T}} P_E = 0$, so $P_{\mathcal{T}} r = P_{\mathcal{T}} v$. For any $z \in \mathcal{T}$,

$$\|r - z\|_{g_0}^2 = \|r - w\|_{g_0}^2 + \|w - z\|_{g_0}^2 ,$$

which proves optimality and uniqueness. Finally, $w, r - w \in E^\perp$ and $r - w \perp \mathcal{T} \ni w$, so all three components are pairwise orthogonal. □

The constrained projection has a small weighted least-squares form. Let $Z : \mathbb{R}^k \to U_R$ have columns spanning $\ker(P_E A_0)$, and set $V = A_0 Z$. Then

$$w = V(V^* V)^\dagger V^* v. \tag{A.9}$$

The adjoint * uses the Euclidean coefficient inner product and g_0 in the prediction space, and † is the Moore–Penrose pseudoinverse. A weighted singular-value decomposition of V shows that Eq. (A.9) is the orthogonal projection onto $\mathrm{Im}(V) = \mathcal{T}$. The resulting response coefficients are stored with each direction for the fixed prediction family.

A.4 UNIQUE WEIGHT SOLUTION AND COORDINATE CONSISTENCY

After directions are fixed, task labels determine their weights through the supervised-loss derivative. Let $B : \mathbb{R}^d \to \mathcal{H}_{\mathcal{S}}$ be the support response operator and let $\mathcal{H}_{\mathcal{S}}$ use the empirical Fisher inner product $\widehat{g}_{\mathcal{S}}$ at q_H. With adjoint B^*, write $G = B^*B$, $c = B^* y_{\mathcal{S}}$, $M = M_{\text{coord}} \succ 0$, and $K = G + M$.

Proposition A.3 (Regularized weight solution)**.** *The unique minimizer of the local objective is $\beta^\star = K^{-1}c$, and*

$$m(\beta^\star) = -\frac{1}{2} c^\top K^{-1} c, \tag{A.10}$$

$$D_\star := c^\top \beta^\star = \beta^{\star\top} K \beta^\star, \qquad \|\beta^\star\|_2 \le \frac{\|c\|_2}{\lambda_{\min}(M)}. \tag{A.11}$$

If $c \neq 0$, then $D_\star > 0$, and $\beta^\star$ is a strict descent direction of $L_{\mathcal{S}} \circ \mathcal{G}_t$ at zero weight.

Proof. Expanding the regularized fitting objective gives

$$\frac{1}{2} \|B\beta - y_{\mathcal{S}}\|^2_{\widehat{g}_{\mathcal{S}}} + \frac{1}{2}\beta^\top M \beta = \frac{1}{2} \|y_{\mathcal{S}}\|^2_{\widehat{g}_{\mathcal{S}}} - c^\top \beta + \frac{1}{2}\beta^\top K \beta.$$

Because $G = B^*B \succeq 0$ and $M \succ 0$, $K \succ 0$. Setting the gradient $K\beta - c$ to zero gives the unique solution, and substitution gives Eq. (A.10). Since $K\beta^\star = c$ and $\lambda_{\min}(K) \ge \lambda_{\min}(M)$, Eq. (A.11) follows. The chain rule gives

$$\frac{\mathrm{d}}{\mathrm{d}\alpha} L_{\mathcal{S}}(\mathcal{G}_t(\alpha\beta^\star))\Big|_{\alpha=0} = -c^\top \beta^\star = -D_\star < 0.$$

The endpoint derivative B is that of the complete adaptation map, so the descent statement is about the final prediction family. □

Here G measures the Fisher cost of prediction changes and c gives their first-order task benefit. The chain rule connects the resulting weights to descent on the complete prediction family. The finite-weight condition is given in Appendix A.6.

Proposition A.4 (Invariance under a coordinate change)**.** *Let $C \in \mathbb{R}^{d\times d}$ be invertible and rewrite the same family as $\beta = C\widetilde{\beta}$. Define*

$$\widetilde{\mathcal{G}}_t(\widetilde{\beta}) = \mathcal{G}_t(C\widetilde{\beta}), \quad \widetilde{B} = BC,$$
$$\widetilde{G} = C^\top G C, \quad \widetilde{c} = C^\top c, \qquad \widetilde{M} = C^\top M C.$$

Then $\widetilde{\beta}^\star = C^{-1}\beta^\star$, and both coordinates produce the same final prediction.

Proof. The transformed normal equations are $C^\top K C\,\widetilde{\beta}^\star = C^\top c$. Multiplying by $C^{-\top}$ gives $KC\widetilde{\beta}^\star = c = K\beta^\star$, hence $C\widetilde{\beta}^\star = \beta^\star$. Substitution into the adaptation map gives the same endpoint. □

Thus direction scaling or a basis change leaves the prediction unchanged when the regularizer is transformed with the coordinates. Scaling directions while keeping the numerical regularizer fixed changes the constraint on update magnitude.

A.5 FROM BASE-POINT DIRECTIONS TO FINITE PREDICTION UPDATES

To track a local direction through the complete adaptation process, we place base and endpoint tangent vectors in a common space. For fixed inputs and weights, let $\mathcal{H}_0 = T_{q_0}\mathcal{Q}_t$ and $\mathcal{H}_H = T_{q_H}\mathcal{Q}_t$, with inner products g_0, g_H. Within a prediction neighborhood where the Fisher metric is smooth and the required logarithms exist, choose a smooth reference path from q_0 to q_H. Parallel transport along this path gives the inner-product-preserving map $\Pi : \mathcal{H}_0 \to \mathcal{H}_H$. Set

$$E_H = \Pi E, \qquad \mathcal{T}_H = \Pi\mathcal{T}, \qquad W_H = \Pi W. \tag{A.12}$$

Then $\mathcal{T}_H \subseteq E_H^{\perp}$ and $\mathrm{Im}(W_H) \subseteq \mathcal{T}_H$. When $q_H = q_0$, take $\Pi = I$; in fixed Euclidean prediction coordinates, Π is also the identity.

Using q_H as base point, write the finite prediction update as

$$\begin{aligned} F(\beta) &= \mathrm{Log}_{q_H} \mathcal{G}_t(\beta), \qquad F(0) = 0, \\ J &= DF(0) = D\mathcal{G}_t(0), \qquad \Delta = J - W_H. \end{aligned} \tag{A.13}$$

Here J is the first-order response of the complete adaptation process and Δ measures its change relative to the base-point directions. Since $D \mathrm{Log}_{q_H}|_{q_H} = I$, the two expressions for J agree. This chart also defines the linearization error: in Euclidean space $F(\beta) = \mathcal{G}_t(\beta) - \mathcal{G}_t(0)$; in a probability space we use the logarithm above.

Proposition A.5 (Finite-update error and subspace deviation)**.** *Suppose F is continuously differentiable on a neighborhood containing $\mathbb{B}_\rho^d = \{\beta : \|\beta\|_2 \le \rho\}$, and*

$$\|DF(\beta) - DF(\gamma)\|_{\mathrm{op}} \le L_{\mathrm{end}} \|\beta - \gamma\|_2 .$$

The output-side operator norm uses g_H. For every $\beta \in \mathbb{B}_\rho^d$,

$$F(\beta) = W_H\beta + \Delta\beta + \varepsilon(\beta),$$

$$\|\varepsilon(\beta)\|_{g_H} \le \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 , \tag{A.14}$$

$$\|F(\beta) - W_H\beta\|_{g_H} \le \|\Delta\|_{\mathrm{op}} \|\beta\|_2 + \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 . \tag{A.15}$$

With $\mathrm{dist}_{g_H}(z, S) = \inf_{s\in S} \|z - s\|_{g_H}$,

$$\|P_{E_H} F(\beta)\|_{g_H} \le \|P_{E_H}\Delta\|_{\mathrm{op}} \|\beta\|_2 + \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 , \tag{A.16}$$

$$\mathrm{dist}_{g_H}(F(\beta), \mathcal{T}_H) \le \|(I - P_{\mathcal{T}_H})\Delta\|_{\mathrm{op}} \|\beta\|_2 + \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 , \tag{A.17}$$

$$\mathrm{dist}_{g_H}(F(\beta), \mathrm{Im}(J)) \le \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 . \tag{A.18}$$

Proof. Integrating along $s\beta$, $s \in [0, 1]$, gives

$$F(\beta) - J\beta = \int_0^1 \big(DF(s\beta) - DF(0)\big)\beta \, \mathrm{d}s.$$

Therefore

$$\|F(\beta) - J\beta\|_{g_H} \le \int_0^1 L_{\mathrm{end}} s \|\beta\|_2^2 \, \mathrm{d}s = \frac{L_{\mathrm{end}}}{2} \|\beta\|_2^2 .$$

Let $\varepsilon(\beta) = F(\beta) - J\beta$ and use $J = W_H + \Delta$ to obtain Eqs. (A.14)–(A.15). Because $P_{E_H} W_H = 0$ and $(I - P_{\mathcal{T}_H})W_H = 0$, applying the two orthogonal projections to the decomposition and using their operator norm at most one gives Eqs. (A.16) and (A.17). Finally, $J\beta \in \mathrm{Im}(J)$, which gives (A.18). □

The finite-update deviation has two sources. Δ describes a first-order change in direction caused by the adaptation process, while $\varepsilon(\beta)$ is the second-order nonlinear term. If $\mathrm{Im}(J) \subseteq \mathcal{T}_H$, the first-order terms in the two subspace bounds vanish. In particular, when $J = W_H$, the deviation from the selected directions is at most second order. All inner products, responses, and projections use the same fixed inputs and weights; they can be reevaluated on another input set with the same definitions.

An elementary map separates the two effects. In Euclidean space, take $E_H = \mathrm{span}\{\boldsymbol{e}_2\}$, $\mathcal{T}_H = \mathrm{span}\{\boldsymbol{e}_1\}$, $W_H\beta = (\beta, 0)$, and

$$F(\beta) = (\beta, a\beta + b\beta^2).$$

Then $\Delta\beta = (0, a\beta)$ and $\varepsilon(\beta) = (0, b\beta^2)$: the first term rotates the local direction and the second curves the finite update. For fixed β and $\alpha\beta \in \mathbb{B}_\rho^d$,

$$\|F(\alpha\beta) - \alpha J\beta\|_{g_H} \le \frac{L_{\mathrm{end}}}{2} \alpha^2 \|\beta\|_2^2 . \tag{A.19}$$

When $J\beta \neq 0$ and $\alpha \neq 0$, the relative error is at most $L_{\mathrm{end}}|\alpha| \|\beta\|_2^2 / (2\|J\beta\|_{g_H})$. Thus reducing the weight controls nonlinear error, while Δ measures consistency between the base-point direction and the endpoint response.

A.6 Loss decrease for finite prediction updates

The paper estimates weights from the endpoint response $B = R_{\mathcal{S}} D\mathcal{G}_t(0)$, so the loss change can be analyzed on the final prediction family. We combine smoothness of the task loss with the nonlinearity of the adaptation map.

On the support prediction space, write $q_{H,\mathcal{S}} = R_{\mathcal{S}} q_H$ and

$$F_{\mathcal{S}}(\beta) = \mathrm{Log}_{q_{H,\mathcal{S}}}(R_{\mathcal{S}} \mathcal{G}_t(\beta)),$$

so $DF_{\mathcal{S}}(0) = B$. Let $\phi(z)$ be the task loss after restoring a prediction with $\mathrm{Exp}_{q_{H,\mathcal{S}}} z$, and define

$$\psi(\beta) = L_{\mathcal{S}}(\mathcal{G}_t(\beta)) = \phi(F_{\mathcal{S}}(\beta)), \qquad h = \nabla\phi(0).$$

The gradient and output norm use $\widehat{g}_{\mathcal{S}}$, so $c = -B^* h$. Assume that $DF_{\mathcal{S}}$ is $L_{\mathrm{end},\mathcal{S}}$-Lipschitz on $\mathbb{B}^d_\rho$, and that $\nabla\phi$ is K_ϕ-Lipschitz on a convex neighborhood containing the image of this ball. Define

$$\begin{aligned} U &= \|B\|_{\mathrm{op}} + L_{\mathrm{end},\mathcal{S}}\rho, \\ H &= \|h\|_{\widehat{g}_{\mathcal{S}}} + K_\phi \left(\|B\|_{\mathrm{op}} \rho + \frac{L_{\mathrm{end},\mathcal{S}}}{2}\rho^2 \right), \\ L_\psi &= K_\phi U^2 + H L_{\mathrm{end},\mathcal{S}}. \end{aligned} \tag{A.20}$$

Proposition A.6 (Finite-step descent condition)**.** *Under these assumptions, $\nabla\psi$ is L_ψ-Lipschitz on $\mathbb{B}^d_\rho$. Let $c \neq 0$, $s_\star = \|\beta^\star\|_2$, and $D_\star = c^\top \beta^\star > 0$. For $0 < \alpha \leq 1$ with $\alpha s_\star \leq \rho$,*

$$\psi(\alpha\beta^\star) - \psi(0) \leq -\alpha D_\star + \frac{L_\psi}{2}\alpha^2 s_\star^2. \tag{A.21}$$

Thus $L_\psi \alpha s_\star^2 \leq D_\star$ implies $\psi(\alpha\beta^\star) - \psi(0) \leq -\alpha D_\star/2 < 0$. For the full weight $\alpha = 1$, a sufficient condition is

$$s_\star \leq \rho, \qquad D_\star \geq L_\psi s_\star^2. \tag{A.22}$$

Proof. The Lipschitz derivative and the integral argument above give, inside the ball,

$$\|DF_{\mathcal{S}}(\beta)\|_{\mathrm{op}} \leq U, \qquad \|F_{\mathcal{S}}(\beta)\|_{\widehat{g}_{\mathcal{S}}} \leq \|B\|_{\mathrm{op}} \rho + \frac{L_{\mathrm{end},\mathcal{S}}}{2}\rho^2.$$

Consequently, $\|\nabla\phi(F_{\mathcal{S}}(\beta))\|_{\widehat{g}_{\mathcal{S}}} \leq H$. Applying the chain rule at two points yields

$$\begin{aligned} \|\nabla\psi(\beta) - \nabla\psi(\gamma)\|_2 &\leq H \|DF_{\mathcal{S}}(\beta) - DF_{\mathcal{S}}(\gamma)\|_{\mathrm{op}} \\ &\quad + U K_\phi \|F_{\mathcal{S}}(\beta) - F_{\mathcal{S}}(\gamma)\|_{\widehat{g}_{\mathcal{S}}} \\ &\leq \left(H L_{\mathrm{end},\mathcal{S}} + K_\phi U^2 \right) \|\beta - \gamma\|_2 . \end{aligned}$$

The last step uses convexity of the ball and the uniform bound U. Since $\nabla\psi(0) = -c$, integrating along $s\alpha\beta^\star$ and applying the descent lemma gives Eq. (A.21). The remaining statements follow by substitution. □

Equation (A.22) connects the local weight solution to the final loss: the label-driven first-order benefit $D_\star$ must exceed the curvature cost of the finite update. Since $D_\star = \beta^{\star\top}(G + M_{\mathrm{coord}})\beta^\star$, the condition $G + M_{\mathrm{coord}} \succeq L_\psi I$ is sufficient when $s_\star \leq \rho$. If $M_{\mathrm{coord}} = \lambda M_0$, with $M_0 \succ 0$, then

$$\lambda\lambda_{\min}(M_0) \geq \max\left\{ L_\psi, \frac{\|c\|_2}{\rho} \right\} \tag{A.23}$$

simultaneously ensures the radius and curvature conditions. The first part follows from Eq. (A.11), and the second from $G \succeq 0$. This gives an explicit descent regime for the full-weight update. The case $\alpha = 1$ is used in the paper, while $0 < \alpha < 1$ traces a scaled path in the same prediction family.

Geometric deviation and task descent can be controlled over the same weight range. If $\mathrm{Im}(J) \subseteq \mathcal{T}_H$ and $L_\psi \alpha s_\star^2 \leq D_\star$, then the finite prediction decreases support loss while its deviations from trajectory orthogonality and conditional reachability are at most $L_{\mathrm{end}}\alpha^2 s_\star^2/2$. In the general case, the corresponding first-order terms in Eqs. (A.16)–(A.17) are added.

A.7 LABEL COMPLEXITY OF A FROZEN PREDICTION FAMILY

Once the prediction family is frozen, target labels select a prediction function in a d-dimensional weight space. We analyze the complete family $\mathcal{G}_t$ with a covering argument.

Let $\mathcal{F}$ denote the resources used to construct and freeze the family. Assume the labeled scenes $(X_i, Y_i)_{i=1}^n$ are independent of $\mathcal{F}$ and are i.i.d. from the target distribution P. Conditional on $\mathcal{F}$, define $\ell_\beta(X, Y) = \ell(\mathcal{G}_t(\beta)(X), Y)$ on the weight ball $\mathbb{B}_R^d$. Assume $0 \le \ell_\beta \le \ell_{\max}$ and

$$|\ell_\beta(X, Y) - \ell_\gamma(X, Y)| \le L_\beta \|\beta - \gamma\|_2$$

uniformly over scenes. The quantities R, $\ell_{\max}$, and L_β are fixed before these labels are used. Let $\mathcal{R}(\beta) = \mathbb{E}_{\mathsf{P}} \ell_\beta(X, Y)$ and $\widehat{\mathcal{R}}_n(\beta) = n^{-1} \sum_i \ell_\beta(X_i, Y_i)$.

Proposition A.7 (Uniform risk bound for the low-dimensional family)**.** *For any fixed $0 < \eta \le R$ and $0 < \delta < 1$, with conditional probability at least $1 - \delta$,*

$$\sup_{\beta \in \mathbb{B}_R^d} \left| \mathcal{R}(\beta) - \widehat{\mathcal{R}}_n(\beta) \right| \le 2L_\beta \eta + \ell_{\max} \sqrt{\frac{d \log(1 + 2R/\eta) + \log(2/\delta)}{2n}}. \tag{A.24}$$

Proof. The Euclidean ball $\mathbb{B}_R^d$ has an η-net $\mathcal{N}_\eta$ of size at most $(1 + 2R/\eta)^d$. This follows by taking a maximal η-separated set: the disjoint balls of radius $\eta/2$ around its points fit inside the ball of radius $R + \eta/2$, and comparing volumes.

For each fixed $\gamma \in \mathcal{N}_\eta$, Hoeffding's inequality applies to the bounded independent loss. A union bound over the net gives, with probability at least $1 - \delta$,

$$\max_{\gamma \in \mathcal{N}_\eta} |\mathcal{R}(\gamma) - \widehat{\mathcal{R}}_n(\gamma)| \le \ell_{\max} \sqrt{\frac{\log(2|\mathcal{N}_\eta|/\delta)}{2n}}.$$

For any β, choose a net point γ within η. The population and empirical risks each change by at most $L_\beta \eta$. The triangle inequality and the net-size bound give Eq. (A.24). □

Taking $\eta = R/\sqrt{n}$ gives

$$\frac{2L_\beta R}{\sqrt{n}} + \ell_{\max} \sqrt{\frac{d \log(1 + 2\sqrt{n}) + \log(2/\delta)}{2n}}.$$

Thus d controls the covering dimension, while R and L_β describe the range and sensitivity of the prediction family. Network size affects these quantities through the constructed family, but it does not enter the covering dimension directly. Because the event holds simultaneously for every weight in the ball, any label-selected $\beta^\star$ in the ball satisfies the same bound. If its empirical loss improves over the zero-weight endpoint by more than twice the bound, its population loss also improves over the zero-weight prediction.

If direction construction uses unlabeled support inputs, condition instead on $(\mathcal{F}, X_1, \ldots, X_n)$ and assume that the labels are conditionally independent. The same proof holds with

$$\mathcal{R}_X(\beta) = \frac{1}{n} \sum_{i=1}^n \mathbb{E}[\ell_\beta(X_i, Y_i) \mid \mathcal{F}, X_1, \ldots, X_n]$$

in place of the population risk. This conditional version describes label estimation on fixed inputs, while Eq. (A.24) uses independent target scenes. In both cases n counts independent scenes; pixels, proposals, and depth locations within one scene remain part of that scene's loss.

These results give the three theoretical links used by SHIFT: the trajectory reference and executable space define candidate directions, endpoint derivatives and finite-update bounds connect direction selection to task loss, and the frozen prediction family restricts target-side function selection to a d-dimensional coordinate space.

B Baseline Relations and Implementation Details

B.1 Shared starting points, native endpoint, and full adaptation

The pair-pretrained checkpoint, local expansion point, and final baseline prediction correspond to different stages of adaptation. Let θ_{pair} denote the shared pair-pretrained checkpoint. It is trained with paired feature supervision from a frozen RGB observer and is the common starting point for the pair-pretrained Head, SSF, LoRA-r8, and Full-FT baselines in Appendix C.1. SHIFT uses the same source model and paired data, while its task-specific architecture, initialization, and native training procedure define the receiver.

Let θ_0 be the point at which the receiver derivative A_0 is evaluated. Together with the optimizer state, task head, and allowed update domain, it determines the subsequent receiver realization. At θ_0, the task prediction is $q_0 = \pi_t \rho_t(\theta_0)$. Let Φ_t be the fixed task update process and $\mathcal{I}_t(\beta)$ the direction injection, with $\mathcal{I}_t(0) = 0$. A parameterized realization has the form

$$\mathcal{G}_t(\beta) = \pi_t \rho_t(\Phi_t(z_0, \mathcal{I}_t(\beta))), \qquad q_H = \mathcal{G}_t(0), \tag{B.1}$$

where z_0 collects the model, optimizer, and task state, and ρ_t reads the prediction model after the update. Zero weights keep the native update and turn off the injection: q_0 is the expansion-point prediction, whereas q_H is the prediction after the native receiver process and is the SHIFT(Native H) endpoint. A function realization directly uses a fixed thermal prediction function as its zero-coordinate endpoint.

The Native-H + Head, SSF, LoRA-r8, and Full-FT baselines in Tables 1–3 instead start from the completed native thermal baseline and apply supervised adaptation with the specified label budget. Native-H is the same baseline denoted by SHIFT(Native H); the suffix identifies the additional adaptation strategy. SHIFT (full) learns combination weights in the frozen family whose zero-weight prediction is q_H.

SHIFT has two levels. The gain from pair-pretrained adapters to SHIFT(Native H) combines task architecture, target initialization, and training budget. The comparison between SHIFT(Native H) and SHIFT (full) fixes $\mathcal{G}_t$ and contrasts zero weights with label-selected weights, measuring the incremental contribution of information selection, weight calibration, and their receiver realization. The reference realization controls in Table 11 vary the implementation rule, while SHIFT(Native H) always denotes the zero-weight endpoint of the fixed $\mathcal{G}_t$.

Table 5: Scope of the comparisons. Gains are computed within the corresponding main evaluation, pooled out-of-fold, or diagnostic setting.

Comparison	Quantity assessed
Pair-pretrained adapters vs. SHIFT(Native H)	Joint effect of architecture, initialization, and training budget
Native-H adapters vs. SHIFT (full)	Supervised adaptation versus calibrated prediction-space transfer
SHIFT(Native H) vs. SHIFT (full)	Incremental transfer benefit under a matched protocol
Transfer-objective OOF	Supervision construction under shared task resources
Matched directions	Direction content under matched fitting capacity
Reference realization	Choice of task-specific implementation
Coupling diagnostics	Finite-response interaction in a separate setting

B.2 Candidate information, response probes, and adaptation dimension

Algorithm 1 uses two kinds of directions. $v_O^{(j)}$ is the change proposed by the observer in the task prediction space, whereas response probes describe changes that the thermal model can produce through an allowed update. For a parameterized realization, let $\mathcal{U}_{\mathrm{allow}}$ be the full allowed parameter domain, and let $D = [\delta_1, \ldots, \delta_k]$ collect selected update probes. The finite update space in the paper is $U_R = \mathrm{Im}(D)$, with response matrix

$$J = [A_0\delta_1, \ldots, A_0\delta_k] = A_0 D. \tag{B.2}$$

Each column is obtained with a JVP, so the Jacobian with respect to all parameters is never stored. Function realizations use the coordinate derivative of the prediction-function composition.

The range, construction signal, and normalization of each probe δ_r are fixed before calibration labels are read. Parameterized probes use fixed cross-modal response signals from the unlabeled paired set $\mathcal{D}_P$ and remain inside the update domains in Table 6; function tasks use fixed thermal prediction-function coordinates. Support labels do not generate or select probes.

The response-column counts 53, 7, 9, and 8 in Table 8 are k, the probe computation size. The ranks of the trajectory space and prediction family are obtained from the corresponding matrices. MSRS uses LoRA and task-head tensor directions, M3FD uses class-conditioned detector responses, monocular depth uses prediction-function coordinates, and stereo depth uses paired-data-induced model responses. The structured construction fixes the computational basis, while linear dependence and the trajectory constraint determine its effective rank.

Table 6: Prediction coordinates, response construction, and probe specification for the four tasks. The count k is independent of the number of updateable parameters.

Task	Prediction coordinates	Response construction	Probe specification	k
MSRS	Nine-class probabilities	LoRA/head tensor JVPs	One paired-distillation update direction per predefined LoRA module/head group	53
M3FD	Objectness, class, normalized box	Class-conditioned detector JVPs	Six foreground-class-conditioned directions plus one background direction; proposal indices are fixed across observer, trajectory, and JVP evaluation	7
MS2 Mono	Scale-quotient log depth	Prediction-function coordinates	Nine fixed thermal prediction functions; each coordinate is the quotient-space log-depth displacement from q_H	9
MS2 Stereo	Metric disparity	Paired-data model responses	One paired disparity-response direction for each of eight fixed recurrent refinement stages	8

All probe definitions, grouping rules, correspondences, and normalization procedures are fixed from the receiver and unlabeled paired data before calibration labels are accessed. Therefore the calibration labels do not change the response basis J; they enter only through c in the final coefficient estimate.

Because the probes span U_R, the implementation uses the same conditional reachable space as the paper:

$$\mathcal{T}_J = \mathrm{Im}(J) \cap E^{\perp} = \mathrm{Im}(A_0) \cap E^{\perp} = \mathcal{T}. \tag{B.3}$$

When U_R is a strict subset of $\mathcal{U}_{\mathrm{allow}}$, this space describes the responses reachable by the selected probes. The full parameter domain specifies updateable modules, the probes specify the computation basis, and the constrained projection selects executable directions.

B.3 Matrix form of the trajectory reference and constrained projection

Stack task tangent vectors over paired samples and valid locations in a fixed order, and let $F \succ 0$ be the Fisher weight matrix, so the inner product is $a^{\top} F b$. The stack is already restricted to valid task coordinates and therefore contains no quotient directions. F includes sample weights, location weights, and the output Fisher metric; continuous blocks use the corresponding weight times Σ^{-1}, and categorical blocks use Eq. (A.1).

Retain $s+1$ native checkpoints $\theta^{(0)}, \ldots, \theta^{(s)}$. Evaluate them on the same inputs, map them to the base point, and form adjacent changes:

$$\begin{aligned} z_a &= Q \operatorname{Log}_{p_0} \rho_t(\theta^{(a)}), \\ Z &= [z_1 - z_0, \ldots, z_s - z_{s-1}], \qquad E = \mathrm{Im}(Z), \\ P_E &= Z(Z^{\top} F Z)^{\dagger} Z^{\top} F. \end{aligned} \tag{B.4}$$

Here † is the Moore–Penrose pseudoinverse, and the trajectory rank is $\mathrm{rank}(F^{1/2}Z)$. Checkpoints are selected at the fixed progress values in Section B.5.

Let N have columns spanning $\ker(Z^\top FJ)$, and set $K = JN$. Then $\mathrm{Im}(K) = \mathcal{T}_J$. For each candidate $v_j = Qv_O^{(j)}$, its projection and implementation coefficients are

$$\begin{aligned} a_j &= (K^\top FK)^\dagger K^\top F v_j, \\ w_j &= Ka_j, \qquad \delta_j^{\mathrm{impl}} = DNa_j. \end{aligned} \tag{B.5}$$

This directly solves the constrained projection: $Z^\top F w_j = 0$, and w_j is reachable within the response space. Since K is orthogonal to E, first removing $P_E v_j$ and then projecting gives the same result. The coefficients δ_j^{impl} are stored with each local direction; a function receiver stores the corresponding function coordinates. Singular values use the truncation rule in Section B.6.

B.4 Endpoint realization and weight learning

The implementation coefficient δ_j^{impl} from Eq. (B.5) both describes the local direction and parameterizes the complete adaptation family. For a parameterized receiver,

$$\mathcal{G}_t(\beta) = \pi_t \rho_t \left(\Phi_t \left(z_0, \sum_{j=1}^{d} \beta_j \delta_j^{\mathrm{impl}} \right) \right). \tag{B.6}$$

The injection location, update data, optimizer state, schedule, and randomness are fixed before calibration, so $\mathcal{G}_t(0) = q_H$. The endpoint response B_j is obtained by a JVP of the same complete map at $\beta = 0$.

The complete update converts local coefficients into a final prediction. Parameterized receivers inject coefficients into parameters or task losses and then run the native optimization process; function receivers combine fixed thermal prediction functions. MSRS uses a LoRA/head update, M3FD keeps its joint detector output with a full-parameter receiver, and stereo depth keeps its matching and recurrent prediction process. The monocular function family can be written as

$$\begin{aligned} \xi_\ell(x) &= \mathrm{Log}_{q_H(x)} q_\ell(x), \\ \mathcal{G}_t(\beta)(x) &= \mathrm{Exp}_{q_H(x)} \left[\sum_{\ell,b} \beta_{\ell b} \phi_{\ell b}(x) \xi_\ell(x) \right], \end{aligned} \tag{B.7}$$

where q_ℓ are fixed thermal prediction functions, $\phi_{\ell b}$ are fixed combination features, and β collects their coefficients.

The target side uses $B_j = R_{\mathcal{S}} D\mathcal{G}_t(0)[e_j]$, rather than treating the local columns w_j as final prediction changes. With B and the support Fisher weight $F_{\mathcal{S}}$ fixed, it solves

$$(B^\top F_{\mathcal{S}} B + M_{\mathrm{coord}})\beta^\star = -\nabla_\beta L_{\mathcal{S}}(\mathcal{G}_t(\beta))|_{\beta=0}. \tag{B.8}$$

M_{coord} controls the combination weights; weight decay on network parameters remains part of the native training process. Under an invertible coordinate change $\beta = C\widetilde{\beta}$, the construction uses WC, $\mathcal{G}_t(C\widetilde{\beta})$, and $C^\top M_{\mathrm{coord}} C$, preserving the prediction family and estimation problem.

B.5 Pair pretraining and native training

Pair pretraining uses uniform four-layer cosine feature distillation. The thermal encoder and a temporary channel projection are updated, while the RGB observer and thermal task head remain fixed. DINOv2/DA3 read blocks 2, 5, 8, and 11; ConvMAE reads four FPN input layers; FastFS reads four encoder outputs. When channels differ, a biased 1×1 convolution maps to the observer dimension. Registered TartanRGBT features are pooled to 8×8 at valid locations, while LLVIP uses global pooling. Layers, images, and valid bins receive equal weight, giving the mean $1 - \cos$ loss. AdamW uses learning rate 10^{-5}, weight decay 0.01, batch size 1, and gradient clipping at 1, and traverses the paired pool once. Dropout is disabled and BatchNorm running statistics are fixed. The exported checkpoint contains the task head and discards the temporary projection.

Table 7: Native thermal training settings. Updates are optimizer steps and Accum. is the gradient accumulation count. Native-training scenes are fixed before all method comparisons and are separated from the calibration support and scoring scenes for the corresponding experiment.

Task	Updates	Microbatch	Accum.	LR	WD
MSRS	4,000	1	4	10^{-4}	0.05
M3FD	4,000	1	4	8×10^{-5}	0.10
MS2 Mono	1,200	2	1	10^{-4}	0.01
MS2 Stereo	1,200	1	2	10^{-4}	0.01

Label isolation. The native training in Table 7 constructs the thermal receiver and native trajectory used as given resources in Section 3.1. Its supervised scenes do not overlap with the calibration support set $\mathcal{S}$ or final evaluation scenes at the scene or sequence level, and the training finishes before support labels are read. After native training, the initialization, trajectory checkpoints, optimizer settings, schedule, and response-construction rules are fixed.

Native training uses the AdamW settings in Table 7, gradient clipping at 1, a linear warmup over the first 5% of updates, and cosine decay to zero. Checkpoints at 0, 25, 50, 75, and 100% of training provide the trajectory reference. MSRS uses pixel cross-entropy, M3FD uses the native RPN/ROI losses, monocular depth uses per-image centered log-depth L1, and stereo uses final-disparity smooth-L1 with $\beta = 1$ px. Depth losses average valid locations within each image and then average images. LoRA-r8 uses $\alpha = 16$, Gaussian initialization with standard deviation 0.02 for A, and zero initialization for B; SSF scale and shift start at 1 and 0. These settings accompany the response-column and endpoint dimensions reported in the paper.

B.6 Geometric statistics, coordinate scales, and evaluation

Geometric statistics use equal sample weight and equal valid-location weight within each sample. MSRS uses a 37×50 probability grid, and MS2 uses a 64×96 coordinate grid. Grid subsampling does not change physical disparity units. Base-point, observer, and trajectory probabilities are lower-bounded by 10^{-6} and renormalized. Monocular depth uses a unit continuous metric in the centered log-depth subspace; stereo uses a unit metric in pixel-disparity coordinates. Detection uses unit covariance for normalized center and log-size boxes. The objectness, conditional-class, and box blocks are averaged separately and weighted by $1/3$; the four box scalars have equal weight.

Gram matrices, SVDs, and low-dimensional solves use float64. Singular values below 10^{-6} times the largest singular value of the decomposed matrix are truncated. Trajectory, response, and constrained-space ranks are computed from their corresponding matrices. Nonzero endpoint responses are normalized with $s_j = \|B_j\|_{\widehat{g}_S}$. With $S = \mathrm{diag}(s_j)$, the coordinate $\gamma = S\beta$ uses unit regularization and a Cholesky solve; the original-coordinate regularizer is $M_{\mathrm{coord}} = S^\top S$. Zero response columns keep zero weight. Gram matrices and column scales use only inputs, predictions, and geometry masks; supervision enters through the loss gradient.

Monocular depth uses per-image median scaling. Stereo depth is converted from disparity with focal length and baseline without median scaling. Predictions are clipped to $[10^{-3}, 80]$ m at evaluation. Stereo EPE and D1 use 256×384 disparity coordinates with a ground-truth range of 0–192 px and a predicted disparity floor of 10^{-3} px. D1 is the percentage of pixels with absolute disparity error above 3 px and relative error above 5%; δ_1 is the fraction with depth error ratio below 1.25. Detection uses score threshold 0.001, class-wise NMS IoU 0.5, and at most 100 detections per image.

Table 8: Response-probe size, trajectory and response ranks, and calibration dimension.

Task	Probe cols. k	$\mathrm{rank}(E)$	$\mathrm{rank}(J)$	$\mathrm{rank}(\mathcal{T})$	Calibration dim. d	Endpoint rank
MSRS	53	4	53	49	9	9
M3FD	7	4	7	3	7	3
MS2 Mono	9	4	9	5	9	5
MS2 Stereo	8	4	8	4	8	4

Table 9: Complete values for Fig. 3. All methods share the base model, readout, update budget, and label pool, and use fixed two-fold pooled out-of-fold evaluation.

Method	MSRS mIoU ↑	M3FD AP ↑	MS2 Mono SI-AbsRel ↓	MS2 Stereo† EPE ↓
SHIFT(Native H)	55.57	29.77	0.1104	0.7030
Direct output KD	58.82	36.11	0.1004	0.6743
Feature KD	52.48	35.02	0.0996	0.6860
EMA consistency	54.06	31.44	0.1069	0.6894
Confidence pseudo-label KD	55.72	34.60	0.0999	0.6797
SHIFT (full)	63.23	38.50	0.0987	0.6666

† Stereo output and pseudo-label KD use a frozen RGB-pretrained FastFS teacher on thermal stereo inputs; Feature KD uses the paired RGB left image.

Table 10: Complete matched-direction results for Table 4. Shuffled and Isometric match direction count, weight learning, and Fisher prediction energy. Mono uses support out-of-fold evaluation; the other tasks use the main evaluation set. D1 is a percentage and δ_1 is a fraction in the stereo panel.

(a) Main evaluation sets

	MSRS	M3FD		
Prediction	mIoU ↑	AP ↑	AP50 ↑	AP75 ↑
SHIFT(Native H)	53.86	29.64	51.02	31.32
Sample-shuffled	55.85	32.03	54.60	33.99
Sign-isometric	55.65	32.59	55.31	34.69
SHIFT (full)	61.30	35.99	60.34	37.67

(b) MS2 Mono: support out-of-fold

Prediction	SI-AbsRel ↓	SI-δ_1 ↑
SHIFT(Native H)	0.0465	0.988872
Sample-shuffled	0.0454	0.989277
Sign-isometric	0.0457	0.989271
Output-only update	0.0453	0.989171
SHIFT (full)	0.0444	0.989547

(c) MS2 Stereo: main evaluation set

Prediction	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
SHIFT(Native H)	0.4023	0.3858	0.0833	0.932846
Sample-shuffled	0.3926	0.3572	0.0819	0.935868
Sign-isometric	0.3936	0.3834	0.0818	0.934326
SHIFT (full)	0.3879	0.3503	0.0807	0.937501

The probe count k controls response computation, whereas the label-side learning dimension is d, determined by the fixed candidate directions and their nonzero endpoint responses. Neither the network parameter count nor the probe count is the calibration dimension.

Table 11: Task-specific receiver controls. Each panel changes one component while retaining the task data, initialization, label budget, direction budget, and evaluation protocol.

(a) MSRS: RGB observer

Configuration	mIoU ↑
Alternative RGB observer	55.96
SHIFT (full)	61.30

(b) M3FD: detector realization

Configuration	AP ↑
Reference realization	32.72
Without incumbent conditioning	30.88
Without receiver projection	35.09
Output-only update	32.12
SHIFT (full)	35.99

(c) MS2 Mono: prediction-function family

Configuration	SI-AbsRel ↓	SI-δ_1 ↑
Base function family	0.1220	0.862920
Alternative scale coordinates	0.1280	0.855055
Fixed weights	0.1225	0.862762
SHIFT (full)	0.1217	0.863359

(d) MS2 Stereo: recurrent realization

Configuration	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
Reference recurrent realization	0.3881	0.3710	0.0809	0.935532
SHIFT (full)	0.3879	0.3503	0.0807	0.937501

Table 12: Receiver update domains and numerical checks. Parameters and tensors describe the update domain; response columns are the probe count k.

Task	Update domain	Parameters	Tensors	Response cols.
MSRS	LoRA/head	1,043,689	102	53
M3FD	Full-FT	104,002,502	254	7
MS2 Mono	Function coordinates	n/a	n/a	9
MS2 Stereo	Full-FT	15,415,241	507	8

Task	Numerical diagnostic	Result
MSRS	LoRA/head response checks	53/53
M3FD	Normal-equation solve error	8.88×10^{-16}
MS2 Mono	Connectivity checks	27/27
	Gram matrix	Full rank
MS2 Stereo	Output orthogonality energy	3.141×10^{-12}

Table 13: Feature–receiver coupling on an independent MS2 Stereo diagnostic set. All configurations reuse the same precomputed perturbation components and update strength; Additive removes the mixed interaction term. D1 is reported as a percentage and δ_1 as a fraction.

Configuration	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
Base	0.1691	0.08	0.0313	0.9993
Feature-only	0.1822	0.28	0.0341	0.9980
Receiver-only	0.1717	0.13	0.0322	0.9988
Additive (no mixed term)	0.1732	0.16	0.0328	0.9980
Full coupled update	0.1641	0.06	0.0311	0.9993

C Additional Adaptation Results

C.1 Paired-pretraining baselines

These comparisons start Head, SSF, LoRA-r8, and Full-FT from a common pair-pretrained checkpoint. The RGB observer supplies uniform paired-feature supervision following cross-modal distillation (Gupta et al., 2016). SHIFT uses the same observer and paired scenes to construct its task-specific receiver and prediction family. These results describe the overall pipelines; the Native-H-initialized comparisons in the main text assess adaptation after the native receiver is already available.

Table 14: MSRS and M3FD results with a 10% thermal-label budget. Pair-pretrained baselines share their paired checkpoint. SHIFT(Native H) is the zero-weight endpoint, and SHIFT (full) uses the matched receiver with calibrated combination weights. All metrics are percentages.

Method	MSRS	M3FD		
	mIoU ↑	AP ↑	AP50 ↑	AP75 ↑
Pair-pretrained baselines				
Head	21.74	12.37	26.41	9.74
SSF	22.16	14.00	29.29	11.34
LoRA-r8	22.64	19.48	37.26	18.58
Full-FT	24.22	24.92	44.19	25.10
SHIFT(Native H)	53.86	29.64	51.02	31.32
SHIFT (full)	61.30	35.99	60.34	37.67

Table 15: MS2 depth results under the same paired-data and label protocol as Table 14. Monocular depth uses per-image median scaling. Stereo EPE and D1 are computed in disparity space, whereas AbsRel and δ_1 use converted metric depth without median scaling. D1 is the percentage of pixels with disparity error above 3 px and relative error above 5%; δ_1 is the fraction with depth error ratio below 1.25.

Method	MS2 Mono		MS2 Stereo			
	SI-AbsRel ↓	SI-δ_1 ↑	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
Pair-pretrained baselines						
Head	0.2090	0.6873	0.7321	2.00	0.1285	0.8576
SSF	0.1712	0.7106	0.6869	1.70	0.1055	0.8820
LoRA-r8	0.1546	0.7547	0.7190	2.06	0.1127	0.8744
Full-FT	0.2463	0.5500	0.5598	1.09	0.0885	0.9138
SHIFT(Native H)	0.1302	0.8253	0.4023	0.39	0.0833	0.9328
SHIFT (full)	0.1217	0.8634	0.3879	0.35	0.0807	0.9375

Table 16: Results with 5 and 20 labeled thermal samples. MSRS and M3FD are percentages; stereo EPE is measured in pixels. SHIFT uses the fixed native endpoint reported above.

Labels	Method	MSRS mIoU ↑	M3FD AP ↑	Mono SI-AbsRel ↓	Stereo EPE ↓
5	Head	10.08	0.01	0.1968	1.0555
	SSF	11.24	0.03	0.1882	0.9853
	LoRA-r8	10.85	0.07	0.1918	1.0575
	Full-FT	11.70	0.11	0.2802	0.8554
	SHIFT (full)	56.47	31.78	0.1296	0.3961
20	Head	16.43	1.59	0.1737	0.8990
	SSF	16.99	1.60	0.1825	0.8389
	LoRA-r8	16.89	1.54	0.2036	0.8773
	Full-FT	17.08	2.42	0.2589	0.6698
	SHIFT (full)	59.36	35.03	0.1246	0.3921

C.2 Complete Native-H adaptation metrics

Tables 17 and 18 report complete metrics for the Native-H-initialized adapters at 5 labels, 20 labels, and the 10% label pool. The main text uses two decimals for recognition scores and D1 and four decimals for depth errors and accuracy fractions. The tables below retain four-decimal D1 values to expose smaller differences.

Table 17: Complete recognition metrics for adaptation from Native-H. All scores are percentages.

Labels	Method	MSRS	M3FD		
		mIoU ↑	AP ↑	AP50 ↑	AP75 ↑
5	Native-H + Head	54.12	29.86	51.31	31.59
	Native-H + SSF	54.27	29.95	51.35	31.59
	Native-H + LoRA-r8	54.29	30.05	51.52	31.67
	Native-H + Full-FT	54.36	30.20	52.05	31.89
20	Native-H + Head	54.78	30.55	52.30	31.99
	Native-H + SSF	55.36	30.66	52.52	32.38
	Native-H + LoRA-r8	55.51	31.56	53.58	33.43
	Native-H + Full-FT	55.60	32.20	55.06	34.05
10%	Native-H + Head	56.21	31.08	53.11	32.66
	Native-H + SSF	56.81	31.73	54.27	33.40
	Native-H + LoRA-r8	57.56	32.82	55.76	34.69
	Native-H + Full-FT	58.33	34.00	57.26	35.68

Table 18: Complete depth metrics for adaptation from Native-H. Mono uses per-image median scaling. Stereo EPE and D1 are disparity metrics; AbsRel and δ_1 use metric depth. Both accuracy columns are fractions, and D1 is a percentage.

Labels	Method	MS2 Mono		MS2 Stereo			
		SI-AbsRel ↓	SI-δ_1 ↑	EPE (px) ↓	D1 (%) ↓	AbsRel ↓	δ_1 ↑
5	Native-H + Head	0.1301	0.8256	0.4016	0.3828	0.0832	0.9331
	Native-H + SSF	0.1300	0.8262	0.4012	0.3841	0.0831	0.9332
	Native-H + LoRA-r8	0.1301	0.8260	0.4014	0.3831	0.0831	0.9332
	Native-H + Full-FT	0.1329	0.8134	0.3995	0.3789	0.0828	0.9337
20	Native-H + Head	0.1281	0.8346	0.4009	0.3805	0.0830	0.9331
	Native-H + SSF	0.1289	0.8322	0.3993	0.3790	0.0828	0.9340
	Native-H + LoRA-r8	0.1295	0.8302	0.3995	0.3792	0.0829	0.9337
	Native-H + Full-FT	0.1322	0.8162	0.3959	0.3698	0.0820	0.9352
10%	Native-H + Head	0.1282	0.8347	0.3985	0.3783	0.0826	0.9339
	Native-H + SSF	0.1264	0.8411	0.3978	0.3729	0.0824	0.9345
	Native-H + LoRA-r8	0.1254	0.8466	0.3959	0.3710	0.0821	0.9346
	Native-H + Full-FT	0.1314	0.8190	0.3918	0.3590	0.0816	0.9359